

PAT-NO: JP410021286A  
DOCUMENT-IDENTIFIER: JP 10021286 A  
TITLE: CIRCUIT SCALE ESTIMATING METHOD  
PUBN-DATE: January 23, 1998

INVENTOR-INFORMATION:  
NAME  
IWASAKI, TOMOE

ASSIGNEE-INFORMATION:  
NAME COUNTRY  
MATSUSHITA ELECTRIC IND CO LTD N/A

APPL-NO: JP08176379  
APPL-DATE: July 5, 1996

INT-CL (IPC): G06F017/50

ABSTRACT:

PROBLEM TO BE SOLVED: To provide the circuit scale estimating method which precisely estimates the circuit scale in consideration of a logical function and common terms of a state machine by inputting a function description.

SOLUTION: This method includes a step 102 wherein the state machine is extracted from the function description and the number of gates of the state machine is estimated, a step 103 wherein the logical function is extracted and the number of gates of the logical function is estimated, a step 104 wherein the number of gates of the remaining part is estimated, and a step 105 wherein the sum of those numbers of gates is regarded as the total number of gates. The logical function is converted into a product sum form and the

state machine

is converted into a state transition table to generate a set of cubes representing product terms. The number of gates of cubes is set according to the dimensions of the cubes, and the multiple estimation of the common terms is prevented by searching a covering cube.

COPYRIGHT: (C)1998,JPO

(19) 日本国特許庁 ( J P )

(12) 公開特許公報 ( A )

(11) 特許出願公開番号

特開平10-21286

(43) 公開日 平成10年(1998) 1月23日

(51) Int.Cl.<sup>6</sup>

G 0 6 F 17/50

識別記号

庁内整理番号

F I

G 0 6 F 15/60

技術表示箇所

6 6 4 Z

6 5 2 E

審査請求 未請求 請求項の数13 O L (全 13 頁)

(21) 出願番号 特願平8-176379

(22) 出願日 平成8年(1996) 7月5日

(71) 出願人 000005821

松下電器産業株式会社

大阪府門真市大字門真1006番地

(72) 発明者 岩崎 知恵

大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

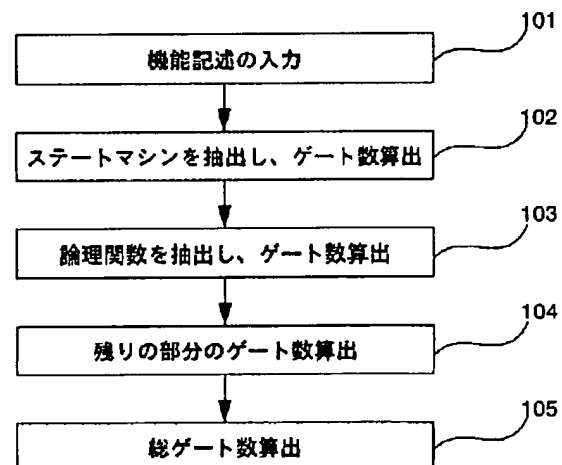
(74) 代理人 弁理士 宮井 暎夫

(54) 【発明の名称】 回路規模見積もり方法

(57) 【要約】

【課題】 機能記述を入力し、論理関数およびステートマシンにおける共通項を考慮して精度よく回路規模を見積もる回路規模見積もり方法を提供する。

【解決手段】 機能記述からステートマシンを抽出しステートマシンのゲート数を見積もるステップ102と、論理関数を抽出し論理関数のゲート数を見積もるステップ103と、残りの部分のゲート数を見積もるステップ104と、それらのゲート数の和を総ゲート数とするステップ105を含む。論理関数は積和形に変換し、ステートマシンは状態遷移表に変換し、積項を表すキューブの集合を作成する。キューブの次元からキューブのゲート数を設定し、カバーするキューブを探すことにより、論理関数の共通項の多重見積もりを防ぐ。



## 【特許請求の範囲】

【請求項1】 大規模集積回路の機能を示すレジスタ転送レベルの機能記述からステートマシンを抽出し、前記ステートマシンを実現する論理回路のゲート数を見積もるステートマシンゲート数見積もり過程と、前記ステートマシンを抽出した前記機能記述の残りの部分のゲート数を見積もる残部ゲート数見積もり過程と、前記ステートマシンゲート数見積もり過程および前記残部ゲート数見積もり過程のゲート数見積もり過程から求めた各ゲート数の和を求めるゲート数合算過程とを含む回路規模見積もり方法。

【請求項2】 大規模集積回路の機能を示すレジスタ転送レベルの機能記述から複数の論理関数を抽出し、前記複数の論理関数を実現する論理回路のゲート数を見積もる論理関数ゲート数見積もり過程と、前記複数の論理関数を抽出した前記機能記述の残りの部分のゲート数を見積もる残部ゲート数見積もり過程と、前記論理関数ゲート数見積もり過程および前記残部ゲート数見積もり過程から求めた各ゲート数の和を求めるゲート数合算過程とを含む回路規模見積もり方法。

【請求項3】 大規模集積回路の機能を示すレジスタ転送レベルの機能記述からステートマシンを抽出し、前記ステートマシンを実現する論理回路のゲート数を見積もるステートマシンゲート数見積もり過程と、前記機能記述から複数の論理関数を抽出し、前記複数の論理関数を実現する論理回路のゲート数を見積もる論理関数ゲート数見積もり過程と、前記ステートマシンおよび前記複数の論理関数を抽出した前記機能記述の残りの部分のゲート数を見積もる残部ゲート数見積もり過程と、前記ステートマシンゲート数見積もり過程、前記論理関数ゲート数見積もり過程および前記残部ゲート数見積もり過程から求めた各ゲート数の和を求めるゲート数合算過程とを含む回路規模見積もり方法。

【請求項4】 ステートマシンゲート数見積もり過程は、ステートマシンを状態遷移表で表し、次の状態のステートレジスタの値または出力信号の値が各々1になる条件を現在の状態の前記ステートレジスタの値および入力信号の値の積和形論理関数で表し、その積項からキューブの集合を作成し、前記キューブの集合のゲート数を算出し、前記キューブの集合のゲート数と前記各積和形論理関数の中の積項の数と前記ステートレジスタの数とから前記ステートマシンのゲート数を求めることを特徴とする請求項1または3記載の回路規模見積もり方法。

【請求項5】 論理関数ゲート数見積もり過程は、複数の論理関数をそれぞれ積和形論理関数に変換し、前記積和形論理関数の積項からキューブの集合を作成し、前記キューブの集合のゲート数を算出し、前記キューブの集合のゲート数と前記各積和形論理関数の中の積項の数から前記複数の論理関数のゲート数を求めることを特徴と

する請求項2または3記載の回路規模見積もり方法。

【請求項6】 状態遷移表で表されたステートマシンのゲート数を見積もる回路規模見積もり方法であって、次の状態のステートレジスタの値または出力信号の値が各々1になる条件を現在の状態の前記ステートレジスタの値および入力信号の値の積和形論理関数で表し、その積項からキューブの集合を作成し、前記キューブの集合のゲート数を算出し、前記キューブの集合のゲート数と前記各積和形論理関数の中の積項の数と前記ステートレジスタの数とから前記ステートマシンのゲート数を求めることを特徴とする回路規模見積もり方法。

【請求項7】 複数の論理関数からなる論理回路のゲート数を見積もる回路規模見積もり方法であって、前記複数の論理関数をそれぞれ積和形論理関数に変換し、前記積和形論理関数の積項からキューブの集合を作成し、前記キューブの集合のゲート数を算出し、前記キューブの集合のゲート数と前記積和形論理関数の中の積項の数から複数の論理関数のゲート数を求めることを特徴とする回路規模見積もり方法。

【請求項8】 キューブの集合で示される論理回路のゲート数を見積もる回路規模見積もり方法であって、前記キューブの中のドントケアの数を前記キューブの次元としたとき、前記キューブのゲート数を変数の数および次元から設定し、ゲート数の降順にソートされた前記各キューブに対して、該キューブをカバーする他のキューブを求め、該キューブをカバーする他のキューブが存在するならば該キューブのゲート数から前記他のキューブのゲート数を引いた値を新しく該キューブのゲート数とし、全キューブのゲート数の和を論理回路のゲート数とすることを特徴とする回路規模見積もり方法。

【請求項9】 キューブの集合の中で各キューブをカバーする他のキューブを求める過程は、前記各キューブの段数の初期値を1とし、前記カバーするキューブの段数とカバーされるキューブの段数に1を足した値とを比較し、前記カバーされるキューブの段数に1を足した値の方が大きくなるときに前記カバーするキューブの段数を前記カバーされるキューブの段数に1を足した値に変更し、前記キューブの段数があらかじめ与えられた段数の制限値以下のキューブに対してのみ前記カバーする他のキューブを求めることを特徴とする請求項8記載の回路規模見積もり方法。

【請求項10】 キューブの集合の中で各キューブをカバーする他のキューブを求める過程を含み、前記キューブの集合で示される論理回路のゲート数を見積もる回路規模見積もり方法であって、前記各キューブの段数の初期値を1とし、前記カバーするキューブの段数とカバーされるキューブの段数に1を足した値とを比較し、前記カバーされるキューブの段数に1を足した値の方が大きくなるときに前記カバーするキューブの段数を前記カバーされるキューブの段数に1

を足した値に変更し、前記キューブの段数があらかじめ与えられた段数の制限値以下のキューブに対してのみ前記カバーする他のキューブを求めることを特徴とする回路規模見積もり方法。

【請求項11】 キューブの集合で示される論理回路のゲート数の見積もり方法であって、前記キューブの中のドントケアの数を前記キューブの次元としたとき、前記キューブのゲート数を変数の数および次元から設定し、ゲート数の降順にソートされた前記各キューブに対して、該キューブをカバーする他のキューブを求め、該キューブをカバーする他のキューブが存在するならば該キューブのゲート数から前記他のキューブのゲート数を引いた値を新しく該キューブのゲート数とし、前記他のキューブが存在しない該キューブを他のキューブの集合に移動した後、前記他のキューブの集合の主項を求めて前記他のキューブの集合にキューブとして追加し、前記他のキューブの集合が空集合となるかまたは主項が求まらなくなるまで前記キューブのゲート数の設定から前記他のキューブの集合の主項を求めて前記他のキューブの集合にキューブとして追加するまでの各処理を繰り返し、前記全てのキューブの集合の全キューブのゲート数の和を論理回路のゲート数とすることを特徴とする回路規模見積もり方法。

【請求項12】 キューブの集合に含まれるキューブの変数の数があらかじめ与えられた制限値以下のときは前記キューブの集合のゲート数を算出し、前記キューブの変数の数が制限値を超えるときは、前記キューブの集合に含まれる各キューブにおける変数間の結合度をもとに前記キューブの変数を2分割し、前記キューブの変数の分割に基づき全てのキューブを2分割して2つのキューブの集合を作成し、前記2つのキューブの集合についてゲート数を算出し、前記キューブの集合の分割を前記キューブの変数の数が制限値を超えなくなるまで繰り返すことを特徴とする請求項8、9、10または11記載の回路規模見積もり方法。

【請求項13】 キューブの集合で示される論理回路のゲート数の見積もる回路規模見積もり方法であって、前記キューブの集合に含まれるキューブの変数の数があらかじめ与えられた制限値以下のときは前記キューブの集合のゲート数を算出し、前記キューブの変数の数が制限値を超えるときは、前記キューブの集合に含まれる各キューブにおける変数間の結合度をもとに前記キューブの変数を2分割し、前記キューブの変数の分割に基づき全てのキューブを2分割して2つのキューブの集合を作成し、前記2つのキューブの集合についてゲート数を算出し、前記キューブの集合の分割を前記キューブの変数の数が制限値を超えなくなるまで繰り返すことを特徴とする回路規模見積もり方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、ハードウェア記述言語で記述された機能記述を用いたLSI（大規模集積回路）設計において、LSIの機能を示すレジスタ転送レベルの機能記述の段階で自動的に、LSIの回路規模を見積もる回路規模見積もり方法に関し、特に機能記述に含まれる論理関数およびステートマシンの回路規模見積もり方法に関するものである。

【0002】

【従来の技術】近年、論理合成ツールの普及により、機能記述を用いたLSI設計が広く行なわれるようになってきた。しかし、人が機能記述を見ただけで回路規模を見積もることは難しく、機能記述に基づく仮論理合成を行ったり、機能記述中の演算子の出現回数から回路規模を見積もっていた。

【0003】前者の仮論理合成による回路規模見積もり方法は、実際の論理合成に用いるコンピュータのソフトウェア処理に類する簡略な論理合成ソフトウェアにより、機能記述を基に論理合成を行ない、所要のゲート数を求めるという方法である。また、後者の機能記述に基づく回路規模見積もり方法は、例えば特開平6-83895号公報に開示されている。この回路規模見積もり方法は、予め、機能記述におけるコンストラクト（if文やcase文などの構成要素）や演算子（+や\*、<、=など）に対してそれが置き換えられるであろうと思われるようなセルを想定し、そのゲート数をライブラリに予め登録しておく。そして、登録したライブラリを用い、機能記述を順次読み取り、コンストラクトや演算子を読み取る毎にライブラリを参照してゲート数を算出し、その合計を回路のゲート数としている。

【0004】上記のゲート数算出の流れ図は図17に示すようになる。図17において、ステップST1では、この機能記述を先頭から行順に読み取っていく。ステップST2では、読み取った内容に応じてゲート数を算出する。ステップST3では、算出したゲート数を記憶する。ステップST4で機能記述の読み取りが終了したかどうかを判断することで、ステップST1～ST3の各ステップを機能記述の最終行まで繰り返し実行する。そして、ステップST5でゲート数のデータを合計して終了する。

【0005】

【発明が解決しようとする課題】しかしながら、前者の方法では、論理合成を行なうため、ゲート数の算出に時間がかかる。また、後者の方法では、機能記述を順に読み込んでいき、各機能記述毎にライブラリを参照してゲート数を算出し、算出した全てのゲート数を加算していくため、ゲート数の合計は、演算子の出現数に依存することとなる。機能記述の中で、コンストラクト（if文やcase文などの構成要素）や算術演算子（“+（加算）”、“-（減算）”など）は機能記述での出現数から求めても誤差は大きくはないが、特に論理演算子

(“&(論理積)”, “|(論理和)”, “!(反転)”など)は論理合成における最適化が考慮されずに共通項を多重見積もりすることになり、誤差が大きくなってしまいう問題点を有していた。例えば、入力A, B, C, Dと出力X, Y, Zとの間について、 $X = A \& B$ ,  $Y = A \& B \& C$ ,  $Z = A \& B \& C \& D$ というような3つの論理関数が機能記述に含まれている場合について考える。1個の“&”を実現するのに1個のゲートが必要であるとする、これら3つの論理関数では“&”が合計6個あるので、ゲート数は6と算出されることになる。しかしながら、実際の論理合成では、上記の3つの論理関数をそのまま論理回路に置き換えるのではなく、例えば共通項(例えば、 $A \& B$ ,  $A \& B \& C$ )は考慮して、ゲート数が最小となるように最適化処理が行われた上で論理回路に置き換えられることになる。この場合、上記の論理関数は、例えば $X = A \& B$ ,  $Y = X \& C$ ,  $Z = Y \& D$ のように変形される。この場合には、“&”が合計3個になり、ゲート数は3個になる。機能記述からの見積もりでは、ゲート数が6個であるのに、論理合成によるゲート数が3個となり、誤差が大きくなる。

【0006】上記のような見積もりの誤差は、論理関数だけでなく、論理関数と同じように論理回路で構成されるステートマシンについても、大きいものとなっていた。本発明の目的は、機能記述の段階で自動的に精度よく回路規模を見積もることができる回路規模見積もり方法を提供することである。本発明の他の目的は、ステートマシンおよび論理関数における共通項の多重見積もりを回避し、機能記述の段階で自動的に精度よく回路規模を見積もることができる回路規模見積もり方法を提供することである。

【0007】

【課題を解決するための手段】請求項1記載の回路規模見積もり方法は、LSIの機能を示すレジスタ転送レベルの機能記述からステートマシンを抽出し、ステートマシンを実現する論理回路のゲート数を見積もるステートマシンゲート数見積もり過程と、ステートマシンを抽出した機能記述の残りの部分のゲート数を見積もる残部ゲート数見積もり過程と、ステートマシンゲート数見積もり過程および残部ゲート数見積もり過程のゲート数見積もり過程から求めた各ゲート数の和を求めるゲート数合算過程とを含む。

【0008】この方法によると、ステートマシンを実現する論理回路のゲート数と、機能記述の残りの部分のゲート数とを分けて見積もり、各ゲート数を合算するので、回路規模を自動的に精度よく見積もることができる。請求項2記載の回路規模見積もり方法は、LSIの機能を示すレジスタ転送レベルの機能記述から複数の論理関数を抽出し、複数の論理関数を実現する論理回路のゲート数を見積もる論理関数ゲート数見積もり過程と、

複数の論理関数を抽出した機能記述の残りの部分のゲート数を見積もる残部ゲート数見積もり過程と、論理関数ゲート数見積もり過程および残部ゲート数見積もり過程から求めた各ゲート数の和を求めるゲート数合算過程とを含む。

【0009】この方法によると、複数の論理関数を実現する論理回路のゲート数と、機能記述の残りの部分のゲート数とを分けて見積もり、各ゲート数を合算するので、回路規模を自動的に精度よく見積もることができる。請求項3記載の回路規模見積もり方法は、LSIの機能を示すレジスタ転送レベルの機能記述からステートマシンを抽出し、ステートマシンを実現する論理回路のゲート数を見積もるステートマシンゲート数見積もり過程と、機能記述から複数の論理関数を抽出し、複数の論理関数を実現する論理回路のゲート数を見積もる論理関数ゲート数見積もり過程と、ステートマシンおよび複数の論理関数を抽出した機能記述の残りの部分のゲート数を見積もる残部ゲート数見積もり過程と、ステートマシンゲート数見積もり過程、論理関数ゲート数見積もり過程および残部ゲート数見積もり過程から求めた各ゲート数の和を求めるゲート数合算過程とを含む。

【0010】この方法によると、ステートマシンを実現する論理回路のゲート数と、複数の論理関数を実現する論理回路のゲート数と、機能記述の残りの部分のゲート数とを分けて見積もり、各ゲート数を合算するので、回路規模を自動的に精度よく見積もることができる。請求項4記載の回路規模見積もり方法は、請求項1または3記載の回路規模見積もり方法において、ステートマシンゲート数見積もり過程は、ステートマシンを状態遷移表で表し、次の状態のステートレジスタの値または出力信号の値が各々1になる条件を現在の状態のステートレジスタの値および入力信号の値の積和形論理関数で表し、その積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と各積和形論理関数の中の積項の数とステートレジスタの数とからステートマシンのゲート数を求めることを特徴とする。

【0011】この方法によると、ステートマシンを状態遷移表で表し、この状態遷移表に基づいてステートマシンを積和形論理関数で表し、積和形論理関数の積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と各積和形論理関数の中の積項の数とステートレジスタの数とからステートマシンのゲート数を求めるので、回路規模を精度良く見積もることができる。なお、キューブの集合のゲート数は論理関数を実現する論理積ゲートの数に対応し、各積和形論理関数の中の積項の数は論理関数を実現する論理和ゲートの数に対応する。

【0012】請求項5記載の回路規模見積もり方法は、請求項2または3記載の回路規模見積もり方法におい

て、論理関数ゲート数見積もり過程は、複数の論理関数をそれぞれ積和形論理関数に変換し、積和形論理関数の積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と各積和形論理関数の中の積項の数から複数の論理関数のゲート数を求めることを特徴とする。

【0013】この方法によると、複数の論理関数をそれぞれ積和形論理関数に変換し、積和形論理関数の積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と各積和形論理関数の中の積項の数から複数の論理関数のゲート数を求めるので、回路規模を精度良く見積もることができる。なお、キューブの集合のゲート数は論理関数を実現する論理積ゲート数に対応し、各積和形論理関数の中の積項の数は論理関数を実現する論理和ゲート数に対応する。

【0014】請求項6記載の回路規模見積もり方法は、状態遷移表で表されたステートマシンのゲート数を見積もる回路規模見積もり方法であって、次の状態のステートレジスタの値または出力信号の値が各々1になる条件を現在の状態のステートレジスタの値および入力信号の値の積和形論理関数で表し、その積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と各積和形論理関数の中の積項の数とステートレジスタの数とからステートマシンのゲート数を求めることを特徴とする。

【0015】この方法によると、ステートマシンを状態遷移表で表し、この状態遷移表に基づいてステートマシンを積和形論理関数で表し、積和形論理関数の積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と各積和形論理関数の中の積項の数とステートレジスタの数とからステートマシンのゲート数を求めるので、回路規模を精度良く見積もることができる。なお、キューブの集合のゲート数は論理関数を実現する論理積ゲートの数に対応し、各積和形論理関数の中の積項の数は論理関数を実現する論理和ゲートの数に対応する。

【0016】請求項7記載の回路規模見積もり方法は、複数の論理関数からなる論理回路のゲート数を見積もる回路規模見積もり方法であって、複数の論理関数をそれぞれ積和形論理関数に変換し、積和形論理関数の積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と積和形論理関数の中の積項の数から複数の論理関数のゲート数を求めることを特徴とする。

【0017】この方法によると、複数の論理関数をそれぞれ積和形論理関数に変換し、積和形論理関数の積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と各積和形論理関数の中の積項の数から複数の論理関数のゲート数を求め

るので、回路規模を精度良く見積もることができる。なお、キューブの集合のゲート数は論理関数を実現する論理積ゲート数に対応し、各積和形論理関数の中の積項の数は論理関数を実現する論理和ゲート数に対応する。

【0018】請求項8記載の回路規模見積もり方法は、キューブの集合で示される論理回路のゲート数を見積もる回路規模見積もり方法であって、キューブの中のドントケアの数をキューブの次元としたとき、キューブのゲート数を変数の数および次元から設定し、ゲート数の降順にソートされた各キューブに対して、該キューブをカバーする他のキューブを求め、該キューブをカバーする他のキューブが存在するならば該キューブのゲート数から他のキューブのゲート数を引いた値を新しく該キューブのゲート数とし、全キューブのゲート数の和を論理回路のゲート数とすることを特徴とする。

【0019】この方法によると、一つのキューブをカバーする他のキューブを求め、他のキューブが存在するときに、一つのキューブのゲート数から他のキューブのゲート数を引いた値を一つのキューブの新しいゲート数とするので、共通項の多重見積もりを回避し、機能記述の段階で自動的に精度よく回路規模を見積もることができる。

【0020】請求項9記載の回路規模見積もり方法は、請求項8記載の回路規模見積もり方法において、キューブの集合の中で各キューブをカバーする他のキューブを求める過程は、各キューブの段数の初期値を1とし、カバーするキューブの段数とカバーされるキューブの段数に1を足した値とを比較し、カバーされるキューブの段数に1を足した値の方が大きくなるときにカバーするキューブの段数をカバーされるキューブの段数に1を足した値に変更し、キューブの段数があらかじめ与えられた段数の制限値以下のキューブに対してのみカバーする他のキューブを求めることを特徴とする。

【0021】この方法によると、キューブの段数を制限することができ、例えばゲートの遅延時間など、信号の伝達タイミングを考慮した上でゲート数を見積もることができる。請求項10記載の回路規模見積もり方法は、キューブの集合の中で各キューブをカバーする他のキューブを求める過程を含み、キューブの集合で示される論理回路のゲート数を見積もる回路規模見積もり方法であって、各キューブの段数の初期値を1とし、カバーするキューブの段数とカバーされるキューブの段数に1を足した値とを比較し、カバーされるキューブの段数に1を足した値の方が大きくなるときにカバーするキューブの段数をカバーされるキューブの段数に1を足した値に変更し、キューブの段数があらかじめ与えられた段数の制限値以下のキューブに対してのみカバーする他のキューブを求めることを特徴とする。

【0022】この方法によると、キューブの段数を制限

することができ、例えばゲートの遅延時間など、信号の伝達タイミングを考慮した上でゲート数を見積もることができる。請求項11記載の回路規模見積もり方法は、キューブの集合で示される論理回路のゲート数の見積もり方法であって、キューブの中のドントケアの数をキューブの次元としたとき、キューブのゲート数を変数の数および次元から設定し、ゲート数の降順にソートされた各キューブに対して、該キューブをカバーする他のキューブを求め、該キューブをカバーする他のキューブが存在するならば該キューブのゲート数から他のキューブのゲート数を引いた値を新しく該キューブのゲート数とし、他のキューブが存在しない該キューブを他のキューブの集合に移動した後、他のキューブの集合の主項を求めて他のキューブの集合にキューブとして追加し、他のキューブの集合が空集合となるかまたは主項が求まらなくなるまでキューブのゲート数の設定から他のキューブの集合の主項を求めて他のキューブの集合にキューブとして追加するまでの各処理を繰り返し、全てのキューブの集合の全キューブのゲート数の和を論理回路のゲート数とすることを特徴とする。

【0023】この方法によると、一つのキューブをカバーする他のキューブを求め、他のキューブが存在するときに、一つのキューブのゲート数から他のキューブのゲート数を引いた値を一つのキューブの新しいゲート数とし、さらに主項を考慮に入れカバーする他のキューブを求めてゲート数を算出するので、共通項の多重見積もりを回避し、機能記述の段階で自動的に精度よく回路規模を見積もることができる。

【0024】請求項12記載の回路規模見積もり方法は、請求項8、9、10または11記載の回路規模見積もり方法において、キューブの集合に含まれるキューブの変数の数があらかじめ与えられた制限値以下のときはキューブの集合のゲート数を算出し、キューブの変数の数が制限値を超えるときは、キューブの集合に含まれる各キューブにおける変数間の結合度をもとにキューブの変数を2分割し、キューブの変数の分割に基づき全てのキューブを2分割して2つのキューブの集合を作成し、2つのキューブの集合についてゲート数を算出し、キューブの集合の分割をキューブの変数の数が制限値を超えなくなるまで繰り返すことを特徴とする。

【0025】この方法によると、キューブを分割してキューブの変数を制限するので、カバーするキューブを求める処理が効率的に行なうことができ、ステートマシンの入力信号の数が多い場合や論理関数を表す変数の数が多い場合でも効率よく回路規模を見積もることができる。請求項13記載の回路規模見積もり方法は、キューブの集合で示される論理回路のゲート数を見積もる回路規模見積もり方法であって、キューブの集合に含まれるキューブの変数の数があらかじめ与えられた制限値以下のときはキューブの集合のゲート数を算出し、キューブ

の変数の数が制限値を超えるときは、キューブの集合に含まれる各キューブにおける変数間の結合度をもとにキューブの変数を2分割し、キューブの変数の分割に基づき全てのキューブを2分割して2つのキューブの集合を作成し、2つのキューブの集合についてゲート数を算出し、キューブの集合の分割をキューブの変数の数が制限値を超えなくなるまで繰り返すことを特徴とする。

【0026】この方法によると、キューブを分割してキューブの変数を制限するので、カバーするキューブを求める処理が効率的に行なうことができ、ステートマシンの入力信号の数が多い場合や論理関数を表す変数の数が多い場合でも効率よく回路規模を見積もることができる。

【0027】

【発明の実施の形態】以下、本発明の実施の形態について、図面を参照しながら説明する。図1は本発明の実施の形態における回路規模見積もり方法の処理を示す流れ図である（請求項1〜3に対応する）。図1を用いて本発明の回路規模見積もり方法における処理の流れを説明する。

【0028】最初にステップ101で、ハードウェア記述言語で記述された機能記述を入力する。つぎにステップ102で、入力した機能記述からステートマシンを抽出しステートマシンのゲート数を算出する。つぎにステップ103で、入力した機能記述から論理関数を抽出し論理関数のゲート数を算出する。つぎにステップ104で、従来技術により機能記述の中の残りの部分のゲート数を算出し、最後にステップ105で、ステップ102、103、104で算出したゲート数の和を求め、それを回路の総ゲート数とする。なお、ステップ102とステップ103の順序は逆であってもよい。また、ステップ102とステップ103の処理は両方実施する必要はなく、何れか片方だけを実施する場合も実施の形態に含まれる。

【0029】以下、ステートマシンおよび論理関数のゲート数見積もりについて詳細に説明する。図2はステップ102におけるステートマシンのゲート数見積もり処理を示す流れ図である（請求項4、6に対応する）。最初にステップ201で、抽出したステートマシンから状態遷移表を作成する。つぎにステップ202で、状態遷移表からキューブの集合を作成する。つぎにステップ203で、キューブのゲート数を算出する。最後にステップ204で、ステートマシンの総ゲート数を算出する。

【0030】図1のステップ102で抽出されるステートマシンは、ステップ201において、入力信号および現在の状態、次の状態、出力信号の関係を表す状態遷移表で表現される。図3(a)は状態遷移表の例であり、同図(b)は同図(a)に対応した状態遷移図である。図3(a)において、1は入力信号I1、I2の値を示す欄、2は現在の状態を表すステートレジスタR1、R



## 11

2の値を示す欄、3は次の状態を表すステートレジスタR1、R2の値を示す欄、4は出力信号O1、O2、O3の値を示す欄である。図3(b)において、各節点にはs0～s2の記号を付してあり、s0はステートレジスタR1、R2の値が“00”の状態を示し、s1はステートレジスタR1、R2の値が“01”の状態を示し、s2はステートレジスタR1、R2の値が“10”の状態を示している。また、矢印付きの枝の近傍には、入力/出力の値を付記している。

【0031】例えば、図3(a)の状態遷移表の1行目は、ステートレジスタの値がR1=0、R2=0で、入力信号の値がI1=0、I2=0の時、次の状態ではステートレジスタの値がR1=0、R2=1になり、出力信号の値がO1=1、O2=0、O3=0であることを示している。状態遷移表においては、入力信号と現在の状態のステートレジスタの値の組み合わせが積和形論理関数における1つの積項となる。

【0032】積和形論理関数において、複数の積項が存在し、積項に含まれる変数の総数をnとすると、各積項は $c \in \{1, 0, x\}$ のn次元のベクトルで表現できる。そのn次元ベクトルをキューブといい、積項に変数が正論理で含まれる場合は“1”で、変数が負論理で含まれる場合は“0”で、積項に含まれていない変数はドントケアを示す“x”で表される。キューブの中の“x”の数をキューブの次元という。上記のキューブは、積和形論理関数の各積項に対応して作成される。

【0033】図4は、図3(a)の状態遷移表から作成されるキューブを示している。図4において、5A、5Bは、状態遷移表の欄3の次の状態のステートレジスタR1、R2の値が各々1になる時の状態遷移表の欄1の入力信号I1、I2と欄2のステートレジスタR1、R2の現在の状態を表すキューブであり、図3(a)の例では各々2個ずつ存在する。6A、6B、6Cは、状態遷移表の欄4の出力信号O1、O2、O3の値が各々1になる時の状態遷移表の欄1の入力信号I1、I2と欄2のステートレジスタR1、R2の現在の状態を表すキューブであり、図3(a)の例では各々1個ずつ存在する。7は図2のステップ202で作成されるキューブ集合であり、各キューブ5A、5B、6A、6B、6Cをまとめたものである。8は各キューブの次元を示している。

【0034】図2のステップ204で求めるステートマシンの総ゲート数は、ステップ203で求められるキューブのゲート数と、ステートレジスタのゲート数と、積項の和演算に必要なORゲートのゲート数と、入力信号の負論理を生成するインバータのゲート数との和である。図4の例では、ステートレジスタR1とR2がそれぞれ2つのキューブすなわち積項を含むため、2つの2入力ORゲートが必要である。

【0035】図5は、図1のステップ103における論

## 12

理関数のゲート数見積もり処理を示す流れ図である(請求項5、7に対応する)。最初にステップ301で、全ての論理関数を積和形論理関数に変換する。つぎにステップ302で、各積項のキューブを作成しキューブ集合を作成する。つぎにステップ303で、キューブのゲート数を算出する。このステップ303は図2のステップ203と同様の処理である。最後にステップ304で、論理関数の総ゲート数を算出する。

【0036】図1のステップ103で抽出される論理関数は、機能記述の中の代入文およびif文やcase文で用いられる判定文の中の、論理演算子と変数が連続した部分あるいは変数と定数との等式の部分である。論理関数は、変数と論理演算子の“&”、“|”、“!”の積和形で表現できる。図5のステップ304で求める論理関数の総ゲート数は、ステップ303で求められるキューブのゲート数と、積項の和演算に必要なORゲートのゲート数と、変数の負論理を生成するインバータのゲート数との和である。

【0037】図6は、論理関数をキューブで表した例である。図6において、9は論理関数、10は論理関数9から得られるキューブ集合、11は各キューブの次元を示している。以下、図2のステップ203および図5のステップ303のキューブのゲート数算出処理の詳細を説明する。

【0038】図7は、キューブのゲート数算出処理を示す第1の実施の形態の流れ図である(請求項8に対応する)。キューブ集合Sの中のキューブの数をN、キューブを構成する変数の数をMとする。最初に、各キューブのゲート数Gcを求める(ステップ401)。キューブのゲート数Gcは、キューブを論理回路として実現するANDゲートの数であり、M-(キューブの次元)-1で求めることができる。つぎに、キューブ集合Sに含まれるN個のキューブをゲート数Gcの降順にソート(並べ替え)する(ステップ402)。つぎに、ソートした順に従い、キューブCi(i=1, 2, ..., N)に対して、キューブCiをカバーする最大ゲート数のキューブCj(i<j)を探す(ステップ403)。もしキューブCiに対してキューブCjが存在すれば、キューブCiのゲート数をGc(i)=Gc(i)-Gc(j)とする(ステップ404)。全てのキューブに対してカバーする他のキューブを探し、ゲート数を変更した後、最後にキューブ集合Sのゲート数を全てのキューブのゲート数の和として求める(ステップ406)。

【0039】以上のように、第1の実施の形態によれば、論理関数やステートマシンをキューブの集合で表し、任意のキューブに対してカバーする他のキューブを求めることにより、論理関数やステートマシンに含まれる共通項の多重見積もりを回避することができ、機能記述の段階で回路規模を自動的に精度よく見積もることができる。

【0040】図8は、キューブのゲート数算出処理を示す第2の実施の形態の流れ図である（請求項9、10に対応する）。図7と同様に、各キューブのゲート数を求め（ステップ401）、キューブをソートする（ステップ402）。さらに、各キューブの段数 $L_c$ に初期値1を設定する（ステップ501）。つぎに、ソートした順に従い、キューブ $C_i$  ( $i=1, 2, \dots, N$ ) に対して、段数 $L_c(i)$ とあらかじめ設定した段数の制限値とを比較し（ステップ502）、段数 $L_c(i)$ が制限値より少なければキューブ $C_i$ をカバーする最大ゲート数のキューブ $C_j$  ( $i < j$ ) を探す（ステップ403）。もし、キューブ $C_i$ に対してキューブ $C_j$ が存在すれば、キューブ $C_i$ のゲート数を $G_c(i) = G_c(i) - G_c(j)$ とし（ステップ404）、キューブ $C_j$ の段数 $L_c(j)$ を、キューブ $C_j$ の段数 $L_c(j)$ と $L_c(i) + 1$ のいずれか大きい方の値とする（ステップ503）。全てのキューブに対してカバーする他のキューブを探し、ゲート数および段数を変更した後、最後にキューブ集合 $S$ のゲート数をキューブのゲート数の和として求める（ステップ406）。

【0041】図10は、図6のキューブ集合10を例として、図7および図8の流れ図を説明するための図である。図10において、12はステップ401でゲート数の降順にソートされたキューブ集合である。13はキューブのゲート数、14は図8の実施の形態の場合のステップ501で設定されるキューブの段数の初期値である。キューブ集合12の1番目のキューブは2番目のキューブに、2番目のキューブは3番目のキューブに、3番目のキューブは4番目のキューブにそれぞれカバー（被覆）されている。15は1, 2, 3番目のキューブに対してステップ404の処理を行なった結果のキューブのゲート数であり、16はステップ503の処理を行なった結果のキューブの段数である。

【0042】以上のように、この第2の実施の形態によれば、論理関数あるいはステートマシンを実現する組み合わせ回路の最大段数を指定することにより、複数の論理関数やステートマシンを含む機能記述からゲート素子の遅延時間など、信号の伝達タイミングを考慮した上で、回路規模を自動的に見積もることができる。その他の効果は、第1の実施の形態と同様である。

【0043】図9は、キューブのゲート数算出処理を示す第3の実施の形態の流れ図である（請求項11に対応する）。図9において、ステップ405までの処理は図7と同様である。1つのキューブ集合に対して、ステップ403からステップ405の繰り返し処理が終了すると、その集合の中でステップ403においてカバーが存在せず変更がなかったキューブを取り出し、他のキューブ集合を作成する（ステップ601）。つぎに、上記他のキューブ集合に対して、主項（prime implicant）を求める（ステップ602）。主項が存在すれば、

主項をその集合に追加し、その集合に対してステップ401からの処理を行なう。以上の処理を他の集合が空集合あるいは主項が存在しなくなるまで繰り返し、最後に全ての集合に含まれる全てのキューブのゲート数の和を求める（ステップ603）。

【0044】図11は、図4のキューブ集合7を例として図9の流れ図を説明するための図である。図11において、17は図4のキューブ集合7に主項としてキューブ18, 19を追加したキューブ集合であり、20はキューブ集合17の各キューブのゲート数を示している。21はキューブ集合17の2番目と3番目のキューブに対してキューブ18をカバーとし、4番目と5番目のキューブに対してキューブ19をカバーとした、ステップ404の後の各キューブのゲート数を示している。

【0045】以上のように、この第3の実施の形態によれば、キューブ集合の中で共通項が容易に見つからない場合に主項を求め集合に追加することにより、複雑な論理関数やステートマシンを含む機能記述から回路規模を自動的に見積もることができる。なお、上記の説明では、キューブのゲート数として求めた値を、総ゲート数の計算に用いるとしているが、3入力以上のANDゲートが使用可能ならば、キューブのゲート数として求めた値+1の数を入力とするANDゲートを想定して総ゲート数の計算に用いてもよい。

【0046】図12(a)は図6の論理関数9を図7の流れ図で求めた場合に想定される論理回路図である。図12(b)は図6の論理関数9を図8の流れ図において段数の制限値を3として求めた場合に想定される論理回路図であり、3入力ANDゲートを使用した回路を想定している。図13は、図11のキューブ11から想定される図3の状態遷移表の論理回路図である。ここでは、3入力および4入力のANDゲートを使用した回路を想定している。

【0047】図14は、キューブのゲート数算出処理を示す第4の実施の形態の流れ図である（請求項12, 13に対応する）。この第4の実施の形態は、大規模なキューブ集合のゲート数見積もりを効率良く行なうために、制限値以下になるまでキューブ集合を分割する前処理を備えた点が前記実施の形態と異なる。この実施の形態では、キューブ集合の2分割を繰り返すためにキューブを使用する。最初に、キューブのゲート数を表すGateを0とする（ステップ701）。つぎに、キューブを構成する変数の数とあらかじめ与えられた制限値とを比較し（ステップ702）、キューブを構成する変数の数が制限値よりも大きければ、変数を2分割する（ステップ703）。つぎに、2分割された変数に従い、キューブ $C_i$  ( $i=1, \dots, N$ ) をキューブ $C_{1i}$ とキューブ $C_{2i}$ とに分割し（ステップ704）、キューブ $C_{1i}$ とキューブ $C_{2i}$ の両方に0または1が存在するキューブ $C_i$ の数を分離部に要するゲート数としてGate

eに加える(ステップ705)。これは、分割した2つのキューブC1i, C2iの積演算を行なうANDゲートに相当する。そして、キューブC1i (i=1, ..., N) からなる集合と、キューブC2i (i=1, ..., N) からなる集合を作成し、キューに追加する(ステップ706)。キューから順に1つずつ集合を取り出し、ステップ702からの処理を行なう(ステップ708)。ステップ702で集合のキューの変数の数が制限値以下ならば、そのキュー集合のゲート数を求め、Gateに加える(ステップ709)。キューが空になれば終了とする(ステップ707)。なお、ステップ709におけるゲート数の算出は、具体的には、先の図7, 図8, 図9に示した方法で行なわれる。

【0048】変数の分割は、キューブ集合の中での変数間の結合度を表したグラフに対して最小カット(min-cut)法を適用する。キューブ集合を構成する変数をグラフの節とし、2つの変数がともに“x”でないキューブの数をその変数に相当する節の間の枝の重みとしてグラフを作成する。最小カット法は、グラフの節を2つの集合SとS' とに分割した場合、集合Sに存在する節と集合S' に存在する節を結ぶ枝の重みが最小となるように集合SとS' を決定する。

【0049】図15は、図14の流れ図を説明するための図である。図15において、22はキューブ集合で、23および24は分割後の2つのキューブ集合である。図16は、キューブ集合22の中での変数間の結合度を表したグラフである。枝に付加した数字がその枝の重みである。図15の例では、図16に示すように分割した場合に、集合SとS' を結ぶ枝の重みの総和が最小となる。集合S、S' の示すキューブ集合は、それぞれ図15の23, 24である。キューブ集合23, 24に含まれる“x”のみのキューブは省略してある。キューブ集合23とキューブ集合24の両方に0または1が存在するキューブは6番目のキューブのみであるため、ステップ705で加えられるゲート数は1である。

【0050】以上のように、この第4の実施の形態によれば、キューブを構成する変数が多い場合に変数間の結合度に基づきキューブを分割しているため、論理関数を表す変数が多い場合、あるいはステートマシンの入力信号の数が多い場合でも、効率よく機能記述から回路規模を見積もることができる。

【0051】

【発明の効果】請求項1記載の回路規模見積もり方法によれば、ステートマシンを実現する論理回路のゲート数と、機能記述の残りの部分のゲート数とを分けて見積もり、各ゲート数を合算するので、回路規模を自動的に精度よく見積もることができ、機能記述の段階で回路規模を精度よく見積もることができ、LSI設計の早い段階でのチップ面積推定が可能となり、設計の効率化を図ることができる。

【0052】請求項2記載の回路規模見積もり方法によれば、複数の論理関数を実現する論理回路のゲート数と、機能記述の残りの部分のゲート数とを分けて見積もり、各ゲート数を合算するので、回路規模を自動的に精度よく見積もることができ、LSI設計の早い段階でのチップ面積推定が可能となり、設計の効率化を図ることができる。

【0053】請求項3記載の回路規模見積もり方法によれば、ステートマシンを実現する論理回路のゲート数と、機能記述の残りの部分のゲート数とを分けて見積もり、各ゲート数を合算するので、回路規模を自動的に精度よく見積もることができ、LSI設計の早い段階でのチップ面積推定が可能となり、設計の効率化を図ることができる。

【0054】請求項4記載の回路規模見積もり方法によれば、ステートマシンを状態遷移表で表し、この状態遷移表に基づいてステートマシンを積和形論理関数で表し、積和形論理関数の積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と各積和形論理関数の中の積項の数とステートレジスタの数とからステートマシンのゲート数を求めるので、論理関数やステートマシンに含まれる共通項の多重見積もりを避けることができ、機能記述の段階で回路規模を精度よく見積もることができる。

【0055】請求項5記載の回路規模見積もり方法によれば、複数の論理関数をそれぞれ積和形論理関数に変換し、積和形論理関数の積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と各積和形論理関数の中の積項の数から複数の論理関数のゲート数を求めるので、論理関数やステートマシンに含まれる共通項の多重見積もりを避けることができ、機能記述の段階で回路規模を精度よく見積もることができる。

【0056】請求項6記載の回路規模見積もり方法によれば、ステートマシンを状態遷移表で表し、この状態遷移表に基づいてステートマシンを積和形論理関数で表し、積和形論理関数の積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と各積和形論理関数の中の積項の数とステートレジスタの数とからステートマシンのゲート数を求めるので、論理関数やステートマシンに含まれる共通項の多重見積もりを避けることができ、機能記述の段階で回路規模を精度よく見積もることができる。

【0057】請求項7記載の回路規模見積もり方法によれば、複数の論理関数をそれぞれ積和形論理関数に変換し、積和形論理関数の積項からキューブの集合を作成し、キューブの集合のゲート数を算出し、キューブの集合のゲート数と各積和形論理関数の中の積項の数から複数の論理関数のゲート数を求めるので、機能記述の段階

で回路規模を精度よく見積もることができる。

【0058】請求項8記載の回路規模見積もり方法によれば、一つのキューブをカバーする他のキューブを求め、他のキューブが存在するときに、一つのキューブのゲート数から他のキューブのゲート数を引いた値を一つのキューブの新しいゲート数とするので、共通項の多重見積もりを回避し、機能記述の段階で自動的に精度よく回路規模を見積もることができる。

【0059】請求項9記載の回路規模見積もり方法によれば、キューブの段数があらかじめ与えられた段数の制限値に達したキューブに対してはカバーを求めない制約を与えることにより、キューブの段数を制限することができ、複数の論理関数やステートマシンを含む機能記述からタイミングを考慮した回路規模を見積もることができる。

【0060】請求項10記載の回路規模見積もり方法によれば、キューブの段数があらかじめ与えられた段数の制限値に達したキューブに対してはカバーを求めない制約を与えることにより、キューブの段数を制限することができ、複数の論理関数やステートマシンを含む機能記述からタイミングを考慮した回路規模を見積もることができる。

【0061】請求項11記載の回路規模見積もり方法によれば、一つのキューブをカバーする他のキューブを求め、他のキューブが存在するときに、一つのキューブのゲート数から他のキューブのゲート数を引いた値を一つのキューブの新しいゲート数とし、さらに主項を考慮に入れカバーする他のキューブを求めてゲート数を算出するので、複雑な論理関数やステートマシンを含む機能記述であっても、共通項の多重見積もりを回避し、機能記述の段階で自動的に精度よく回路規模を見積もることができる。

【0062】請求項12記載の回路規模見積もり方法によれば、キューブの変数の数があらかじめ与えられた制限値を越える場合に、全てのキューブを2分割したキューブの集合を作成する手段を設けることにより、論理関数を表す変数が多い場合、あるいはステートマシンの入力信号の数が多く場合でも、効率よく機能記述から回路規模を見積もることができる。

【0063】請求項13記載の回路規模見積もり方法によれば、キューブの変数の数があらかじめ与えられた制限値を越える場合に、全てのキューブを2分割したキューブの集合を作成し、キューブを分割してキューブの変数を制限するので、カバーするキューブを求める処理が効率よく行なうことができ、ステートマシンの入力信号の数が多く場合や論理関数を表す変数の数が多く場合でも効率よく回路規模を見積もることができる。

【図面の簡単な説明】

【図1】本発明の実施の形態における回路規模見積もり方法の処理を示す流れ図である。

【図2】ステートマシンのゲート数見積もり処理を示す流れ図である。

【図3】(a)は状態遷移表の例であり、(b)は同図(a)に対応する状態遷移図である。

【図4】図3の状態遷移表から作成されるキューブを示す概略図である。

【図5】論理関数のゲート数見積もり処理を示す流れ図である。

【図6】論理関数をキューブで表した例を示す概略図である。

【図7】キューブのゲート数算出処理を示す第1の実施の形態の流れ図である。

【図8】キューブのゲート数算出処理を示す第2の実施の形態の流れ図である。

【図9】キューブのゲート数算出処理を示す第3の実施の形態の流れ図である。

【図10】ゲート数算出処理の第1および第2の実施の形態を説明するための概略図である。

【図11】ゲート数算出処理の第3の実施例を説明するための概略図である。

【図12】(a)は図6の論理関数を図7の流れ図で求めた場合に想定される論理回路図、(b)図6の論理関数を図8の流れ図で求めた場合に想定される論理回路図である。

【図13】図3の状態遷移表を図9の流れ図で求めた場合に想定される論理回路図である。

【図14】キューブのゲート数算出処理を示す第4の実施の形態の流れ図である。

【図15】ゲート数算出処理の第4の実施の形態を説明するための概略図である。

【図16】キューブ集合の中の変数間の結合度を表したグラフである。

【図17】従来例の回路規模見積もり方法を示す流れ図である。

【符号の説明】

102 ステートマシンのゲート数を算出するステップ

103 論理関数のゲート数を算出するステップ

5A, 5B ステートレジスタの値を1にするキューブ

6A, 6B, 6C 出力信号の値を1にするキューブ

7, 10, 12, 17, 22 キューブ集合

8, 11 キューブの次元

13, 20 キューブのゲート数の初期値

14 キューブの段数の初期値

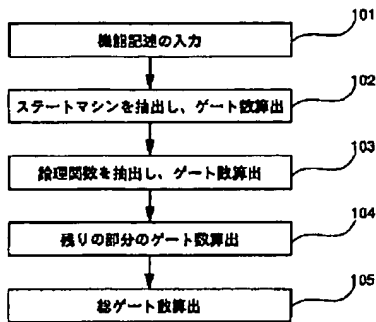
15, 21 キューブの最終ゲート数

16 キューブの最終段数

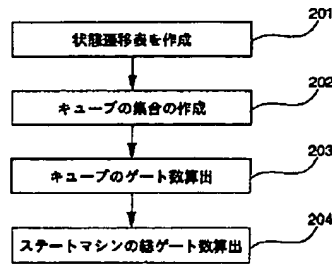
18, 19 主項

23, 24 2分割したキューブ集合

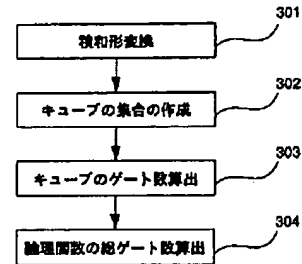
【図1】



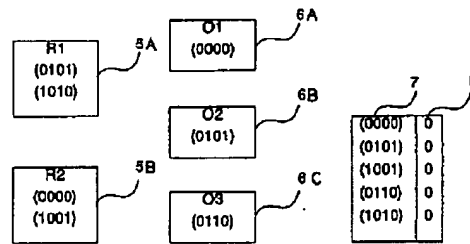
【図2】



【図5】



【図4】

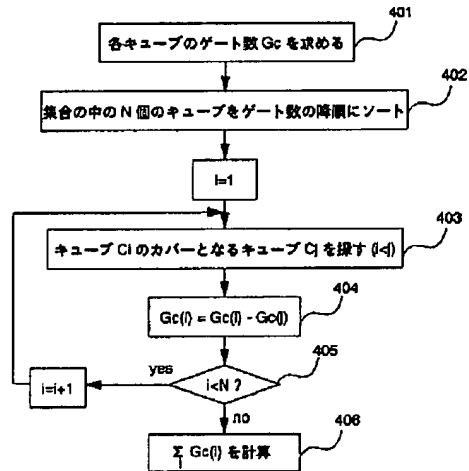


【図3】

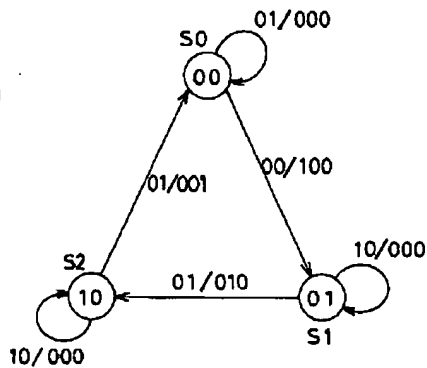
(a)

1	2	3	4
入力信号 I1 I2	現在の状態 R1 R2	次の状態 R1 R2	出力信号 O1 O2 O3
0 0	0 0	0 1	1 0 0
0 1	0 0	0 0	0 0 0
0 1	0 1	1 0	0 1 0
1 0	0 1	0 1	0 0 0
0 1	1 0	0 0	0 0 1
1 0	1 0	1 0	0 0 0

【図7】



(b)

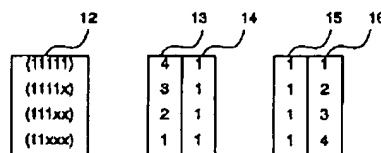


【図6】

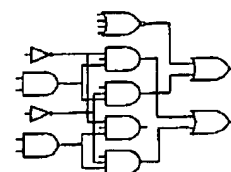
(b)

10	11
OUT1 = A & B	(11xxx) 3
OUT2 = A & B & C	(111xx) 2
OUT3 = A & B & C & D	(1111x) 1
OUT4 = A & B & C & D & E	(11111) 0

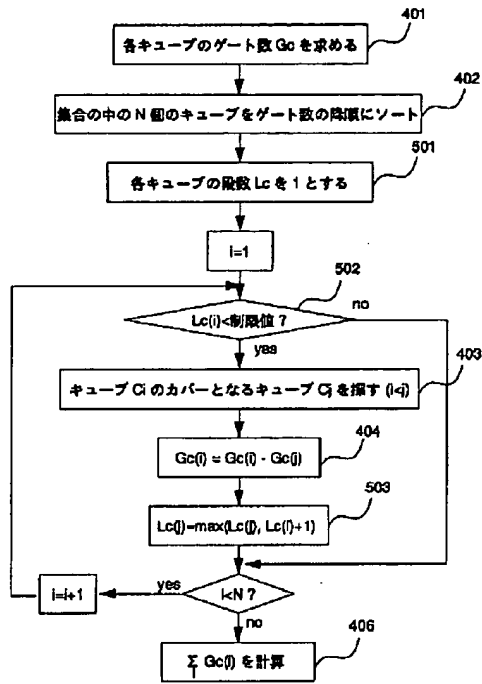
【図10】



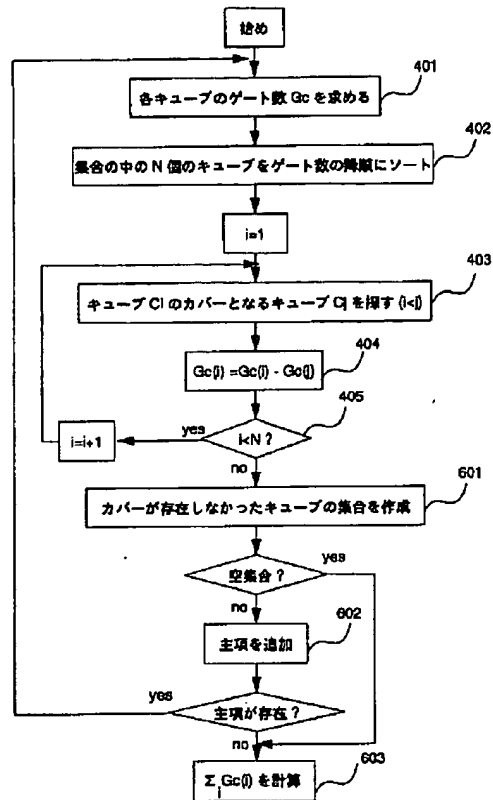
【図13】



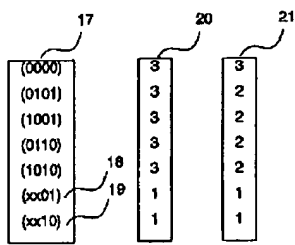
【図8】



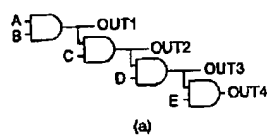
【図9】



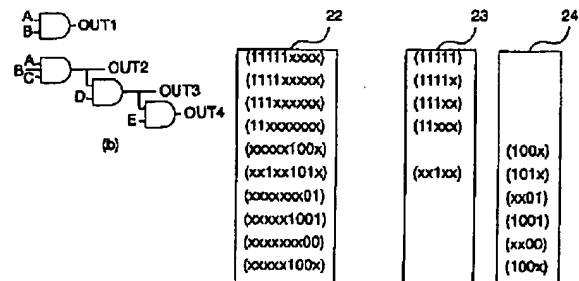
【図11】



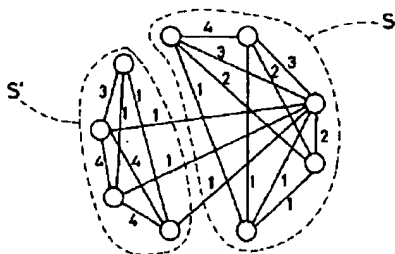
【図12】



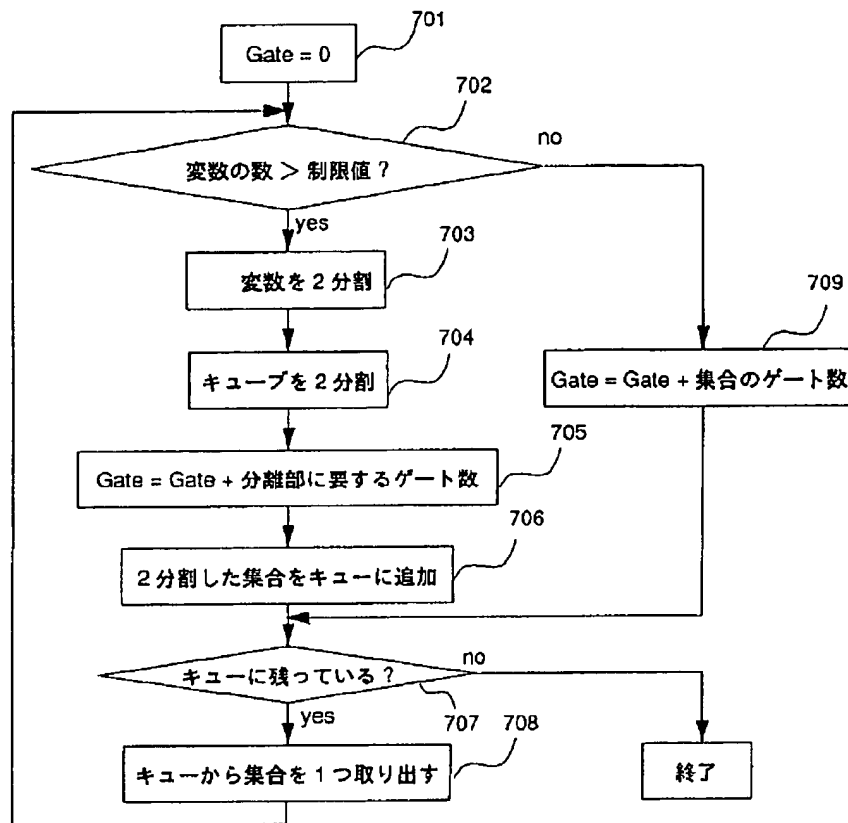
【図15】



【図16】



【図14】



【図17】

